Co-evolution Transformer for Protein Contact Prediction (Appendix)

A Societal Impacts

The proposed method is dedicated to protein contact prediction, which is an essential building block of many important protein structure related tasks. We believe this study will benefit to future research in the structural biology field and contribute to the understanding of protein functions. This study is an application of computational biology and the experiments are conducted without using any living organisms. It is difficult to measure its directly negative societal impact on biology. Algorithms might be developed on the method presented in this paper, and applying these derived algorithms themselves might have negative societal impacts.

B Additional Experiment Details

B.1 Data

Dataset All the models are trained on the structures extracted from PDB (version 2020-03) [1]. We filter the domains with 100% sequence identity, which leads to a dataset consisting of 96,167 non-redundant training domains, denoted as PDB100. To study the effect of training data, we also filter the domains with 30% sequence identity to obtain a smaller training set consisting of 31,344 non-redundant domains, denoted as PDB30. All the data used in this study are derived from the public databases and do not contain any personally identifiable information or offensive content. The data splits of training/validation sets are attached in the supplementary materials.

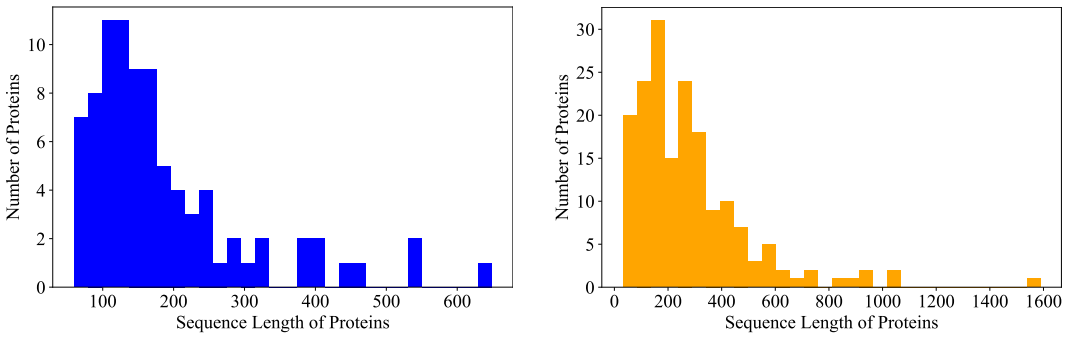
Benchmark Two canonical benchmarks, i.e., CASP14 and CAMEO, are used for the model evaluation, where the CASP14 targets are available at https://predictioncenter.org/ and the CAMEO targets are available at https://www.cameo3d.org/. As shown in Figure 1, both CASP14 and CAMEO include proteins of lengths in a wide range, showing the representativeness and rationality of the two benchmarks. Our method outperforms the baselines significantly on these proteins.

MSA Generation CoT is trained on the protein sequences with each one has 12 MSAs, and one out of the 12 MSAs is sampled for the sequence each time during the training phrase. We generate the MSAs by searching four databases with two open-source tools as follows. HHblits (version 3.3.0) [2] is used to search UniRef30 (version 2020-02) and BFD30 (version 2019-03) [3], while HMMER (version 3.3.2) [4] is used to search UniRef90 (version 2020-02) [5] and MGnify90 (version 2019-05) [6]. For each combination, we run the tool three times with different E-values, i.e., 1, 0.1, and 0.001, obtaining 12 MSAs in total for each protein sequence.

License of Assets HHblits and HMMER are free softwares distributed under the GNU General Public License terms and the 3-Clause BSD open source license terms, respectively.

MSA Representation Inspired by [7], we represent one MSA as a collection of sequence pairs, where each pair consists of the target protein and another homologous protein. The element of each position in the paired sequence is encoded by a binary vector of 41 dimension. The first 20 dimension

35th Conference on Neural Information Processing Systems (NeurIPS 2021).



(a) CASP14 Domains (b) CAEMO Targets

Figure 1: The protein sequence length distributions

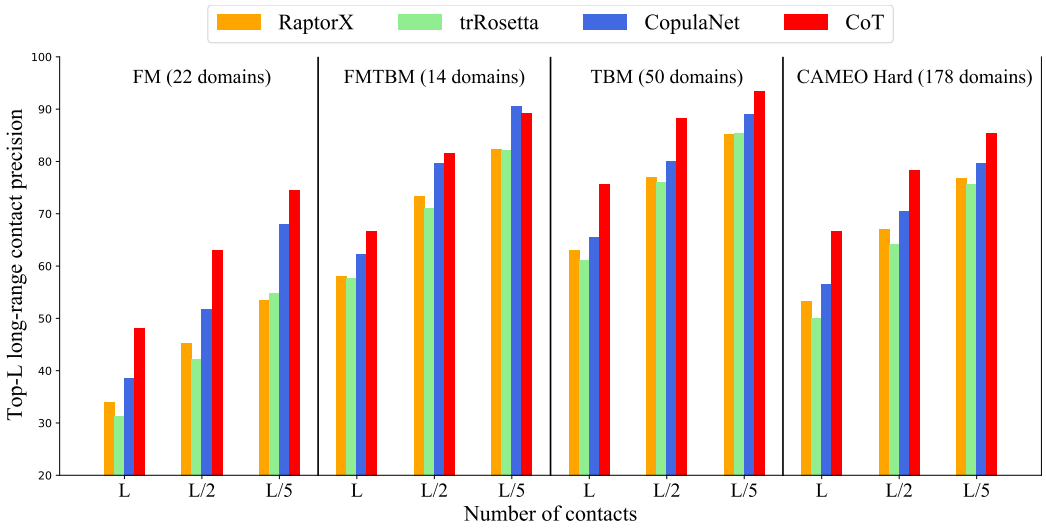


Figure 2: Comparison with SOTA baselines on CASP14 and CAMEO

values are the one-hot representation of the target protein residue and the other 21 dimension values are the one-hot representation of the homologous protein residue. The additional dimension in the latter one is used for the gaps in the homologous proteins represented by a special character '-'.

B.2 Experimental Evaluation

Implementation of Baselines The predictions of top-3 groups in the CASP14 challenge were download from https://predictioncenter.org/. For the three SOTA methods, RaptorX [8], trRosetta [9], and CopulaNet [7], we download their model checkpoints from https://github.com/j3xugit/RaptorX-3DModeling/, https://github.com/gjoni/trRosetta/ and https://github.com/fusong-ju/ProFOLD/ respectively without any further retraining, and then evaluate their models and CoT on the identical MSAs, i.e., MSAs searched from BFD30 with E-value 0.001. For the sake of fairness, we evaluate all predictions with exactly the same evaluation scripts.

The Setting of CoT We summarize the main hyperparameters as follows:

- Optimizer: Adam [10] (learning rate $= 4 \times 10^{-4}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$);
- Weight decay: 10^{-4};
- Nonlinearity: ReLU;
- Warmup: $10,000$ steps;
- A Cosine learning rate scheduler;

Table 1: Ablations for CoT on CASP14 FM domains. *CNN Layers* refers to the number of ResNet blocks in the *co-evolution enhancement* submodule. The bolded line refers to the configuration and result for the final CoT model.

CoT Layers	Embedding Size	CNN Layers	Parameters	*Precision*@L
2	128	32	14.4M	44.7
4	128	16	15.2M	46.8
6	**128**	**12**	**16.6M**	**48.2**
6	256	12	24.9M	48.3

- Training cost: about 30 hours on 4 Tesla V100 GPU cards for $100,000$ steps.

Comparison with MSA Transformer As discussed in Section 4.2, MSA Transformer [11] wasn't included in the comparison for two reasons: 1) it is an unsupervised model pretrained on a large amount of extra sequence data; 2) we cannot access its supervised fine-tuning code/script in the public repository. Therefore, we cannot reproduce the results on the CASP14 benchmark. Nevertheless, we rerun the CoT model on the CASP13-FM dataset to compare with it, obtaining a 65.0% *Precision*@L score, which is better than 57.1% reported in MSA Transformer. For a fair comparison, we have filtered protein sequences similar to CASP13 targets (sequence identity more than 25%) in the training set.

Ablation Study Comprehensive comparisons on two benchmarks are summarized in Figure 2. CoT achieves consistently superior performance on all the kinds of proteins.

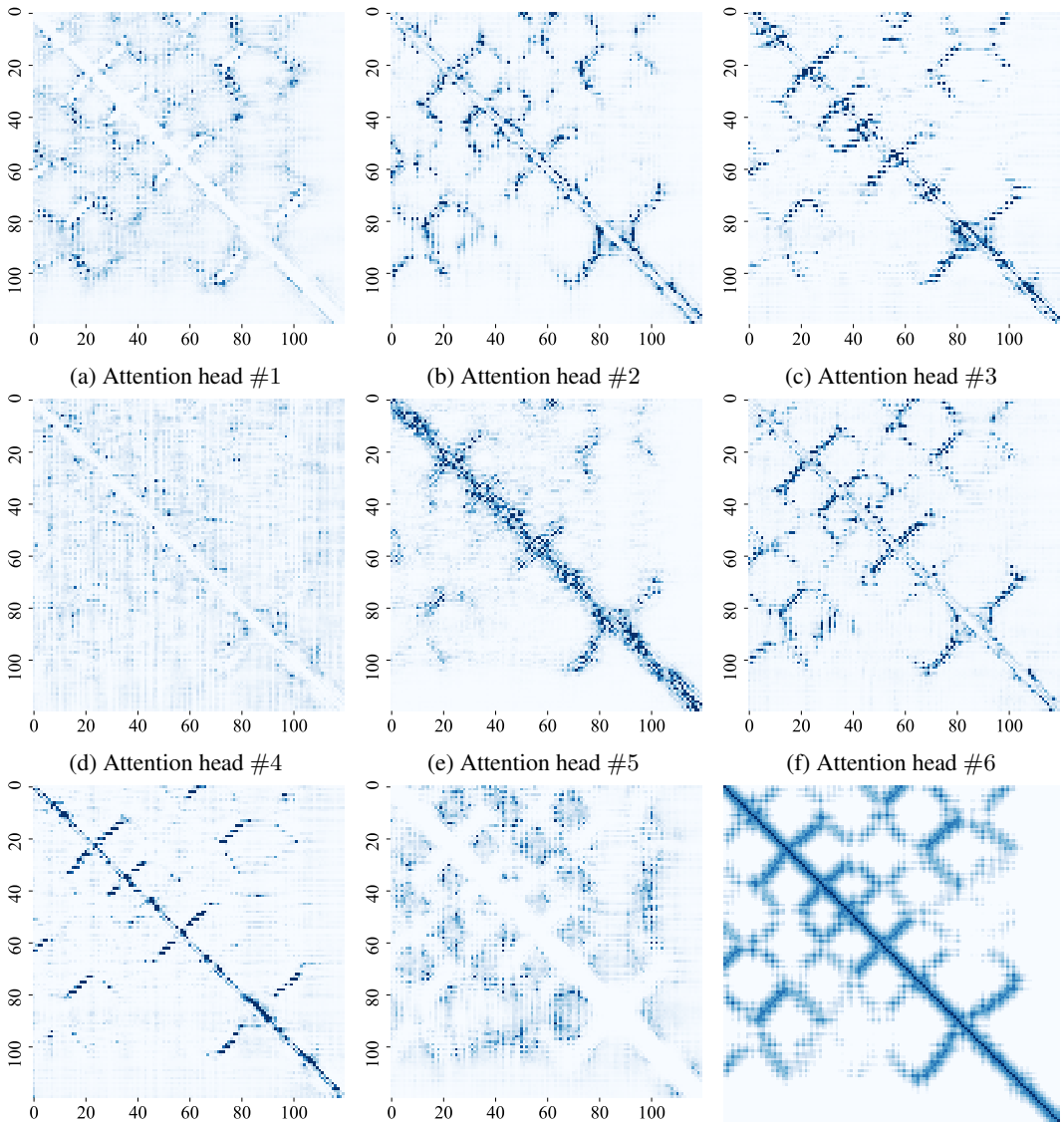
To further study the effect of the other hyperparameters on the model performance, we conducted several ablative experiments, and the results are shown in Table 1. In order to validate the effectiveness of the proposed co-evolution Transformer architecture, we implement two model variants which are equipped with less CoT layers (i.e., 2 and 4) but have comparable parameters with the original CoT model, which is achieved by increasing the depth of CNN in the *co-evolution enhancement* submodule. The result shows that increasing CoT layers can continuously improve the performance of PCP. We also train another model with larger model capacity by increasing embedding size from 128 to 256. The resulting model achieves a higher *Precision*@L score than before, demonstrating that model capacity is also an important factor in this task.

To study the attention patterns that the co-evolution attention module has learned, we visualize the values from all the attention heads of the final layer of CoT. As shown in Figure 3, most attention heads reflect a close relationship with the ground-truth contact map, especially for the 2-nd, 3-rd, and 6-th attention heads, which demonstrates that CoA is capable of extracting contact related patterns.

Statistical Analysis for CASP14-FM/TBM Domains In practice, we notice that the improvement of the CoT model over the CopulaNet model on CASP14-FM/TBM domains varies a lot, and a possible reason is that the performance varies a lot on the small number of domains. To further validate the effectiveness of CoT, we conduct a statistical test on these domains. Regarding that 14 FM/TBM samples are not enough to derive a statistical significance, we re-evaluated the CoT model and the CopulaNet on FM/TBM domains with 12 groups of MSAs (described in Section 4.1), obtaining 168 *Precision*@L scores for each method. The average *Precision*@L scores of CoT and CopulaNet are 60.9% and 55.9%, respectively. Assume the results follow a normal distribution, we conduct a Paired-samples t-test for the two groups of results, obtaining a t-statistic score 5.4 with p-value 2.31×10^{-7}. Since the results may be not normally distributed, the Wilcoxon signed-rank test is further conducted, obtaining a t-statistic score $2,713.0$ with p-value 4.34×10^{-9}. These two results demonstrate that CoT is better than CopulaNet significantly in statistics.

The Impact of the MSA Depth As discussed in Section 4.5, the quality of predicted contacts is highly correlated to M_{eff}. To study the correlation, we plot the values of *Precision*@L and M_{eff} on the CASP14 FM domains as shown in Figure 4.

Performance on different MSA generation pipelines As discussed in Section 4.2, different methods usually adopt different MSA generation pipelines during the evaluation phrase. To further



(a) Attention head #1 (b) Attention head #2 (c) Attention head #3

(d) Attention head #4 (e) Attention head #5 (f) Attention head #6

(g) Attention head #7 (h) Attention head #8 (i) Ground-truth Contacts

Figure 3: Co-evolution attention maps extracted from the final layer of CoT for `4q2z_H`

explore the impact of generation pipelines on the performance of our model, we re-evaluate the CoT model on the four MSAs generated from different databases. The empirical results are summarized in Table 2.

The performance of Co-evolution Transformer for different MSAs varies a lot. For FM domains, CoT performs the best on the BFD30 generated MSA. For the CAMEO targets, CoT has similar performance scores on the first three generated MSAs. A possible reason is that, the FM domains are harder than the CAMEO targets due to the fact that fewer homologs for FM domains exist in most databases (e.g., UniRef). Therefore, CoT is able to obtain a better result on an extremely large database BFD30 for FM domains.

We also conduct the same experiments for the CopulaNet model, which shows a similar phenomenon as CoT does.

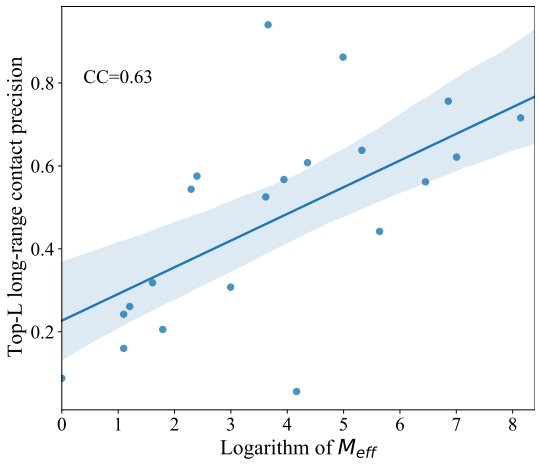


Figure 4: Correlation between *Precision*@L and M_{eff} on the CASP14 FM domains. CC refers to the correlation coefficient.

Table 2: Comparisons for four different MSA generation pipelines on CASP14 and CAMEO (*Precision*@L)

Method	Benchmark	UniRef30	UniRef90	BFD30	MGnify90
CoT (ours)	CASP14-FM	29.6	27.6	48.1	39.1
	CAMEO	67.6	66.6	66.6	58.1
CopulaNet	CASP14-FM	25.1	25.4	38.5	31.8
	CAMEO	58.0	57.0	56.5	49.2

References

[1] Stephen K Burley, Helen M Berman, Charmi Bhikadiya, Chunxiao Bi, Li Chen, Luigi Di Costanzo, Cole Christie, Ken Dalenberg, Jose M Duarte, Shuchismita Dutta, et al. RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic acids research*, 47(D1):D464–D474, 2019.

[2] Michael Remmert, Andreas Biegert, Andreas Hauser, and Johannes Söding. HHblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nature methods*, 9(2):173–175, 2012.

[3] Martin Steinegger, Milot Mirdita, and Johannes Söding. Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nature methods*, 16(7):603–606, 2019.

[4] Robert D Finn, Jody Clements, and Sean R Eddy. HMMER web server: interactive sequence similarity searching. *Nucleic acids research*, 39(suppl_2):W29–W37, 2011.

[5] Baris E Suzek, Hongzhan Huang, Peter McGarvey, Raja Mazumder, and Cathy H Wu. UniRef: comprehensive and non-redundant uniprot reference clusters. *Bioinformatics*, 23(10):1282–1288, 2007.

[6] Alex L Mitchell, Alexandre Almeida, Martin Beracochea, Miguel Boland, Josephine Burgin, Guy Cochrane, Michael R Crusoe, Varsha Kale, Simon C Potter, Lorna J Richardson, et al. MGnify: the microbiome analysis resource in 2020. *Nucleic acids research*, 48(D1):D570–D578, 2020.

[7] Fusong Ju, Jianwei Zhu, Bin Shao, Lupeng Kong, Tie-Yan Liu, Wei-Mou Zheng, and Dongbo Bu. CopulaNet: Learning residue co-evolution directly from multiple sequence alignment for protein structure prediction. *Nature Communications*, 12(1):1–9, 2021.

[8] Jinbo Xu. Distance-based protein folding powered by deep learning. *Proceedings of the National Academy of Sciences*, 116(34):16856–16865, 2019.

[9] Jianyi Yang, Ivan Anishchenko, Hahnbeom Park, Zhenling Peng, Sergey Ovchinnikov, and David Baker. Improved protein structure prediction using predicted interresidue orientations. *Proceedings of the National Academy of Sciences*, 117(3):1496–1503, 2020.

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] Roshan Rao, Jason Liu, Robert Verkuil, Joshua Meier, John F Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. MSA transformer. *bioRxiv*, 2021.